Postal

NWG/RFC # 151

NIC # 6755

A. Shoshani
SDC
May 10, 1971

COMMENTS ON A PROFERRED OFFICIAL ICP
(RFCs 123, 127)

Bob Long at SDC noticed that the order in which messages go out to the network depends on the local NCP. In particular, commands may be given priority over data and therefore, in the sequence specified for the server in RFC 123 (top of Page 3), the last two INIT commands may go out before the data = S on socket = L is sent. (This is the case in the current implementation of SDC's NCP.) The implication is that the user's NCP should be prepared to keep the INIT's it received from the server until the user process gets the data = S, and issues two INITs in response.

This case is brought up now so that people will think about it before the Atlantic City meeting, and comment whether their NCP can tolerate it. It may be necessary to make it explicit in the ICP that the two INITs sent by the server should go out only after the data = S is sent, or even after the user process acknowleges its receipt.

I have a more general remark about the ICP. This is a third level protocol, and therefore should not alter or ignore procedures of the second level protocol (Host-Host protocol). In particular, three remarks seem appropriate:

1. In RFC 123 (bottom of Page 2) it is suggested that the byte size for the connection to the server socket L is 32. However, in the modifications to second level protocol (RFC 107) it is specified that it is up to the sending process to chose the byte size. According to the Host-Host protocol, NCPs should be prepared to accept messages in any byte size ($1 \leq$ size ≤ 255); therefore there is no need to impose a size of 32 in this case. Furthermore, since it is up to the sender to choose the byte size, some Hosts may choose a particular byte size (for simplicity and convenience) and their NCP may not be geared to transmit in an imposed byte size.

You lose

2. In RFCs 66 and 80, an ALL is expected on the connection to the server socket before data can be sent. In RFCs 123 and 127 the ALL requirement disappeared. But the ALL is a Host-Host protocol requirement and not requiring it creates a special case. A particular NCP implementation may cause the ALL to be sent internally when a connection is created, without the user process having control of it. Relaxing this requirement will create a special case for the receiving NCP not to send the ALL and for the sending NCP to send the data = S without first receiving an ALL.

Typo

3. In RFC 127, I disagree with the comment "send 32 bits of data in one message" because it is a second level protocol decision that a message can be sent in any size pieces and the size is to be specified through the ALL mechanism. In particular, there may be hosts which are not prepared to accept more than few bytes at a time (TIPs).

You lose

In general we should not make second level decisions in a third level protocol.